International Journal of Computer Science Engineering and Information Technology Research (IJCSEITR)
ISSN(P): 2249-6831 ISSN(E): 2249-7943
Vol. 4, Issue 5, Oct 2014, 31-36

TRANS STELLAR
• Journal Publications • Research Consultancy

EFFICIENT VIRTUAL MACHINES MIGRATION IN CLOUD COMPUTING

JASPREET KAUR[1], MANPREET KAUR[2] & SAHIL VASHIST[3]

[1,2]Research Scholar, Department of Computer Science and Engineering, Chandigarh Engineering College, Mohali, Punjab, India

[3]Assistant Professor, Department of Computer Science and Engineering, Chandigarh Engineering College, Mohali, Punjab, India

ABSTRACT

Cloud Computing means Internet computing where virtual servers provides platform, software, infrastructure to clients on utility basis. Virtualization is the component of cloud computing. Virtualization is a wide term and it means to create virtual version of something. It may be software environment, hardware, network or storage. Virtual machine migration means to move the virtual machine running on one physical host to another. When the original virtual machine is running and the migration process of that virtual machine is in progress, is known as live migration. This paper focus on maximize the cloud profit and efficiency by efficient placement of virtual machines. The target of proposed algorithm is to maximize the service providers in the case of current resources are not enough to process all the requests on time. In this work, the ranking of request is done on the basis of the profits they can bring. Simulation results show the efficiency of our framework.

KEYWORDS: Efficiency, Migration, Response Time, Cloud Service Providers (CSPs)

INTRODUCTION

Cloud computing offers “computing as a service” model to the subscribers on utility basis [1] [2]. It offers software, platform and infrastructure as a service [3] [4]. Virtualization technique means to run multiple operating systems at the same time on single physical host. It provides support for the creation of customizable, isolated and secure environment for the execution of applications. Virtualization offers opportunity to create scalable systems, which provide additional power at minimal cost. Therefore, it is used highly to offer elastically scalable environment on demand. Virtualization is used widely in modern IT datacenters. Virtualization has three major components: host, virtualization layer and guest. The guest is a system component which communicates with virtualization layer. The host component indicate original environment, it provide support to the guest [5]. The virtualization layer provides environment for the operation of guest by replicating the different or same environment. Some of the characteristics of virtualization are given below:

- **Portability**

According to the particular type of virtualization, the portability acts in different manners. In hardware virtualization, the guest is packed as virtual image and can be migrated and executed on different VMs. In case of programming level virtualization, implemented by JVM and.NET runtime, can be executed without recompiling on corresponding VM.

- **Security**

The power to handle the execution of a guest environment in an absolutely transparent way opens new doors for offering controlled and secure execution environment. The virtual machine means an emulated environment where the guest executes. The guest activities are controlled by the VMM (virtual machine manager). Resources offered by the host computer are simply abstracted from the guest. Sensitive information on the guest is also protected or simply hidden.

- **Monitored Execution**

Execution environment virtualization provides security as well as other features such as isolation, emulation, sharing and aggregation.

Note that in this paper we will use CSPs and VMs as abbreviations of Cloud service providers and virtual machines respectively.

LITERATURE SURVEY

Virtualization facilitates migration of VM from source to destination. VM migration is very useful for data centers administrator and clusters, it provide clear separation between software and hardware. Virtual machine migration offers load balancing, efficient resource usage and energy saving. Methods to migrate virtual machines are divided into two parts: live migration and non live migration. In case of live migration, while migrating, the virtual machine on source keep running, maintain the status. Consumers do not feel any type of interruption in live migration. The state of a VM to migrate is transferred. State means local file system and memory contents.

There is no need to transfer local file system. First of all, virtual machine is suspended, and then state is transferred, and finally, virtual machine is resumed at the target host. Nelson et al. [6] described the implementation and design of a system which make the use of VM technology to offer quick, transparent application movement, neither the data nor the OS required to be changed. Performance is calculated with 100 VMs, migrating at the same time with standard organization benchmark.

It indicates that the application downtime due to movement is reduced to seconds; also time for various workloads is reduced. And introduced live migration technique which is implemented in VMw are and XEN virtualization platform. Zhao et al. [7] presented the resume/suspend migration performance of VMs rather than the live migration. Evaluation focuses on the virtual machine based resource reservation anomalies such as reservation of CPU, network and memory resources for each Virtual machine instance and also for virtual machine cluster. The scenario of vacating one or more physical servers to start the cluster of virtual machines for running parallel processes is also considered. To achieve the goal of time, a model is presented that can characterize the virtual machine migration process and estimate its performance on the basis of experimental analysis. Wood et al. [8] implemented a system having named Sandpiper by the use of live migration technique to remove hotspots in data centers. Sandpiper system automates the detection and monitoring of hotspots, present a mapping of physical resources to virtual resources and start the necessary migration.

Sandpiper implements two different approaches that are black box and gray box. Former is fully OS and application agnostic and later explain OS and application level statistics. The whole system is implemented on XEN and detailed evaluation is conducted by using CPUs, memory intensive applications and network. Single server hotspots are

resolved within 20 seconds. Bobroff et al. [9] proposed the generic algorithm. The system constructed can compute the impact of each scheduling scheme and select the minimum cost one. This solution takes the cost not the revenue into account, which is more important. Steinder et al. [10] presented a scheduling algorithm using Pareto Optimality. The target is to maximize the revenue of the service providers and also minimize the cost of the users. But this is still a theoretical model.

SYSTEM MODEL

In the proposed framework we have used the efficient virtual machine migration algorithm. As shown in figure 1 we can see that broker [11] has managed to get all the required information of other CSPs. By getting information from other CSPs we mean the free VMs which the broker can lend from the other datacenter within or outside CSPs as shown in figure 1. The VMs details are being offered by the other CSPs on regular basis and each broker is capable to view and compare all the virtual machines from its and other sources. If we look at the algorithm as shown below broker is responsible for maintaining all the virtual machines.

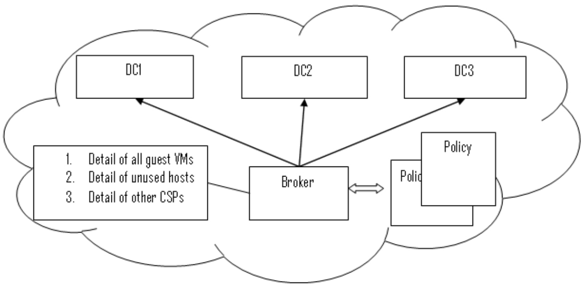

Figure 1: Schematic Model

Each virtual machine is tagged by the health card which checks the capability of virtual machines for a particular load. Only after this checking the virtual machines are allocated to the users following steps of efficient virtual machine Algorithm has been used to carry out this research work.

Efficient virtual machine migration Algorithm for placement of VM works in the following manner:

```
1.  Let N := total number of VM in Cloud Service Providers (CSPs)
    Let M := total number of VM currently under use
2.  Let x  := N-M  //total number of unused guest VMs
3.  Let page[N] := set of all guest VM pages
4.  Let pivot := 0; bubble := 0  // Initialization
5.  ActivePush (Guest VM)   // task monitoring
6.  While bubble < max (pivot, N-pivot) do
7.  Get_health_card(vmID); // report card or health card
8.  Vm_in_Queue  for transmission
9.  bubble++
10. PageFault (Guest-page X)
11. Discard pending queue
```

Figure 2: Pseudo Code for Efficient Virtual Machine Migration

SIMULATION RESULTS

In our proposed framework, we have used CloudSim [12] to implement efficient VM migration algorithm. In Table 1 we can see that the minimum and maximum time taken by our proposed algorithm.

Table 1: Time Taken by our Proposed Algorithm

User Base	Average (ms)	Minimum (ms)	Maximum (ms)
UB1	299.774	246.117	363.138
UB2	301.239	241.636	370.614
UB3	299.31	240.191	369.141
UB4	300.161	234.14	370.613
UB5	300.135	229.619	373.666

Above table shows the average, minimum and maximum value of response time according to the location of the user. Userbase means the location of the user, which is taken into consideration while allocating virtual machine to the cloudlets. Virtual machine is allocated to the user according to the distance of user from datacenter.

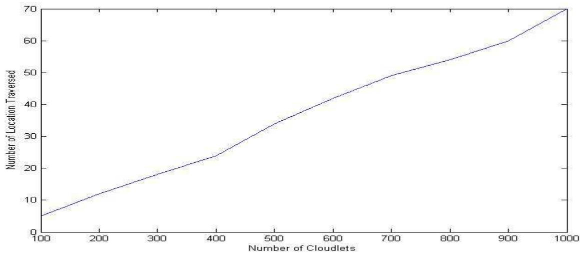

Figure 3: Location Traversed

In the results we can observe that we have number of cloudlets requests and the number of virtual machine migration takes place (traversing). In the figure 3, "70" times the location has been changed by the virtual machines.

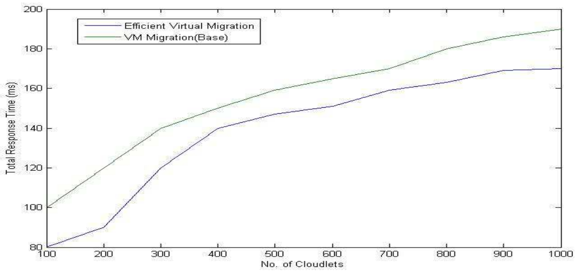

Figure 4: Efficient VM Migration versus Base

And in the figure 4 we can noticed that efficient virtual machine migration uses its own virtual machines and other CSP's VMs due to which it enhance its performance and decreases its response time as compared to our base algorithm [13]. It is shown in the table 2 as below:

Table 2: Comparison Table

No. of Cloudlets	VM Migration(Base)	Efficient VM Migration
	Total Response Time(Ms)	Total Response Time(Ms)
100	100	80
200	120	90
300	140	120
400	150	140
500	160	145
600	165	150
700	170	160
800	180	165
900	185	170
1000	190	170

CONCLUSIONS AND FUTURE SCOPE

Our system is implemented successfully. In this paper the important role of virtualization and migration are

discussed. And the drawbacks of existing system are described. In this paper we have evaluated the response time and cost of the proposed frame work and found that the efficiency of the proposed framework is superior than base paper. Further we can add the parameter of SLA violation and will try to minimize it. The effect of minimization could be very crucial in overall performance increasing and managing virtual machines.

REFERENCES

1. M. Armbrust *et al*, (2010), “A View of Cloud Computing,” *Commun. ACM*, vol. 53, no. 4, pp. 50–58.
2. Michael A, Armando F, Griffith, R, Anthony D. J. et al, “Above the Clouds: A View of Cloud Computing”.
3. Kaur, M. and Vashist, S. (2014), “A review of DOS-DDOS attack and their prevention mechanisms in cloud computing”, *International journal of Computer Application and Technology (IJCAT),* Vol. 1, No. 1, pp. 4-7.
4. Kaur, J. and Vashist, S. (2014), “Third party Auditor in cloud computing using RSA and AES algorithm”, *International journal of computer application and technology IJCAT*, Vol. 1, No. 1, pp. 1-3
5. Agarwal, A. and Raina, S. (2012), “Live Migration of Virtual Machines in Cloud”, *International Journal of Scientific and Research Publication*, Vol. 2, No. 6, pp 1-5.
6. Nelson M, Lim B. and Hutchins G. (2005), “Fast transparent migration for virtual machines,” in *Proceedings of the annual conference on USENIX Annual Technical Conference*.
7. Zhao, M. and Figueiredo, R. (2007), “Experimental study of virtual machine migration in support of reservation of cluster resources,” in *Proceedings of the 3rd international workshop on Virtualization technology in distributed computing*.
8. Wood, T, Shenoy, P. and Venkataramani A. (2007), “Black-box and gray-box strategies for virtual machine migration,” in *Proc. Networked Systems Design and Implementation*.
9. Bobroff, N, Kochut, A, Beaty, K. (2007), "Dynamic Placement of Virtual Machines for Managing SLA Violations", *10th IFIP/IEEE International Symposium on Integrated Network Management*, pp.119-128.
10. Steinder, M, Whalley, I, Carrera, D, et al, (2007), "Server virtualization in autonomic management of heterogeneous workloads," *10th IFIP/IEEE International Symposium on Integrated Network Management.*, pp.139-148
11. Vashist, S. and Singh, R. (2013), “Energy Cost and QoS based management of Data Centers Resources in Cloud Computing”, *International Journal of Computer Science & Engineering Technology (IJCSET),* Vol. 4, No. 6, pp. 663-669.
12. Buyya, R, Ranjan, R, and Calheiros, R. N. (2009), “Modeling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities”, Proc. *of the 7th High Performance Computing and Simulation Conference (HPCS09), IEEE Computer Society.*
13. Mishra, M, Das, A, Kulkarni, P, Sahoo, A, (2012), "Dynamic resource management using virtual machine migrations", *Communications Magazine, IEEE*, Vol. 50, No. 9, pp. 34-40